саме сателіт знаходиться до нього найближче. Практично це реалізовано так, що при надходженні запиту від клієнта до сателіту, nginx отримує наступні дані про клієнта:

•*IP*-адреса клієнта (в даному випадку унікальний ідентифікатор клієнта. У випадку якщо 2 клієнти мають однаковий IP-адрес вважаємо, що це один і той самий клієнт).

Nginx передає ці дані на програмну реалізацію сателіта.

У програмній реалізації клієнта на основі отриманих даних можна отримати шляхи від себе і кожного із сателітів до клієнта (*BGP-PATH*). Тобто, отримати список усіх можливих шляхів від клієнта до потрібної інформації.

Також практично можна визначити:

•*RTT* (*Round-trip delay time*) - час від моменту посилки запиту до моменту отримання відповіді;

•Географічне місце положення клієнта маючи *GEOIP* (Рис. 3).

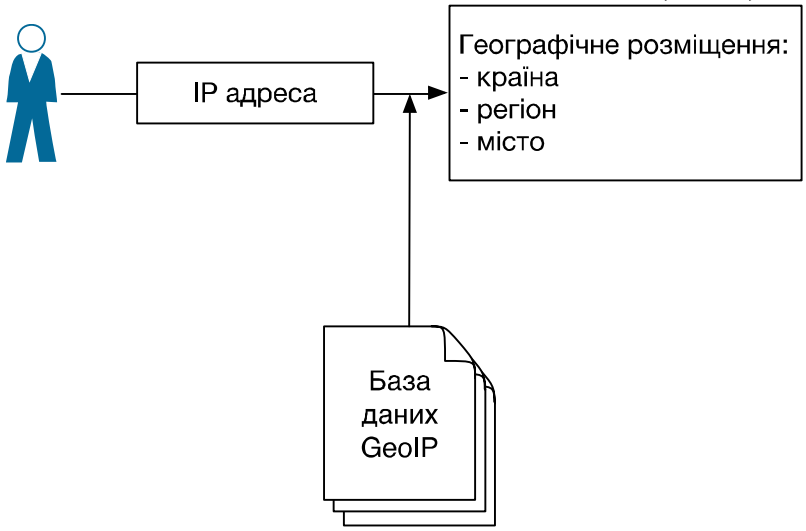

Рис. 3. Метод визначення географічного розміщення користувача

На основі географічного розміщення користувача достатньо легко обчислити адміністративну відстань від нього до усіх можливих сателітів. Звичайно, досить точно цю відстань визначити неможливо, але для практичних потреб оптимальності трафіку підійде й наближене значення. Було б добре, щоб ця відстань вимірювалась хопами, що не завжди вдається практично через обмеженості деяких сегментів мережі.

Можливі два варіанти обчислення відстані між вузлом клієнта і сателітом:

•За різницею *IP* адрес. Тобто, чим більше біт в адресі зверху є однаковими, тим ближче адреса. Такий підхід дозволяє досить швидко і без додаткових обчислювальних затрат виконати цю дію. Адже цікавить не конкретна відстань, а найкоротша з відомих.

•Визначення відстані за допомогою *ICMP*. Але цей метод має недолік - коли на шляху проходження трафіку заборонений трафік *ICMP*, що цілком можливе для специфічних хмаркових сховищ даних.

Однак, визначення найближчого за адміністративною відстанню сателіта до клієнта не завжди гарантує найкращу швидкість обміну даними. Тому важливо ще отримати час від моменту посилки запиту до моменту отримання відповіді від сателіта до клієнта.

Однак, отримати цей параметр можна лише в тому випадку якщо клієнт заходив коли небуть на відповідний сателіт і ця інформація була збережена. Тому потрібно вести статистику відвідування клієнтів з наступними даними:

•*IP*-адрес клієнта;

•*IP*-адрес сателіту;

•*BGP-PATH*;

•*RTT*.

Крім того потрібно враховувати завантаженість сателітів на відповідний час роботи з клієнтами.

Усе це оптимально теоретично і практично працює при наявності достатньої кількості статистичних даних у базі.

При відсутності статистичних даних, на основі яких можна прийняти рішення про оптимальний шлях від клієнта до сателіта, доводиться приймати рішення в умовах невизначеності. Невизначеність у даному випадку достатньо погано формалізується, тому й було прийняте рішення використання практичного підходу.

Невизначеність проявляється у неможливості точно оцінити шлях до конкретного клієнта від сателітів. Тому варто оцінювати цей шлях за іншою доступною інформацією.

•У базі з часом накопичується інформація про різних клієнтів (різні IP). Тому завжди існує можливість отримати відстань до конкретного, навіть якщо він не поточний. І оцінювати відстань від поточного клієнта до потенційно близьких за допомогою визначення відстані за різницею *IP* (а це достатньо проста логічна процедура). Відсортована вибірка з бази даних клієнтів за IP дозволить провести ранжування клієнтів за близькістю до поточного.

•Якщо клієнт знаходиться досить близько (*GEOIP*) до іншого клієнта, то їхні шляхи до відповідного сателіта (*BGP-PATH)* з достатнім наближенням можна вважати однаковими і, відповідно, їхні *RTT* до цього сателіту приблизно рівними.

•Маючи ці дані можна визначити мінімальний *RTT* від клієнта до сателіту. На основі цього відбувається підключення клієнта до його інформаційного сховища через мінімальний шлях, перенаправивши його на ближчий і не завантажений на даний момент сателі.

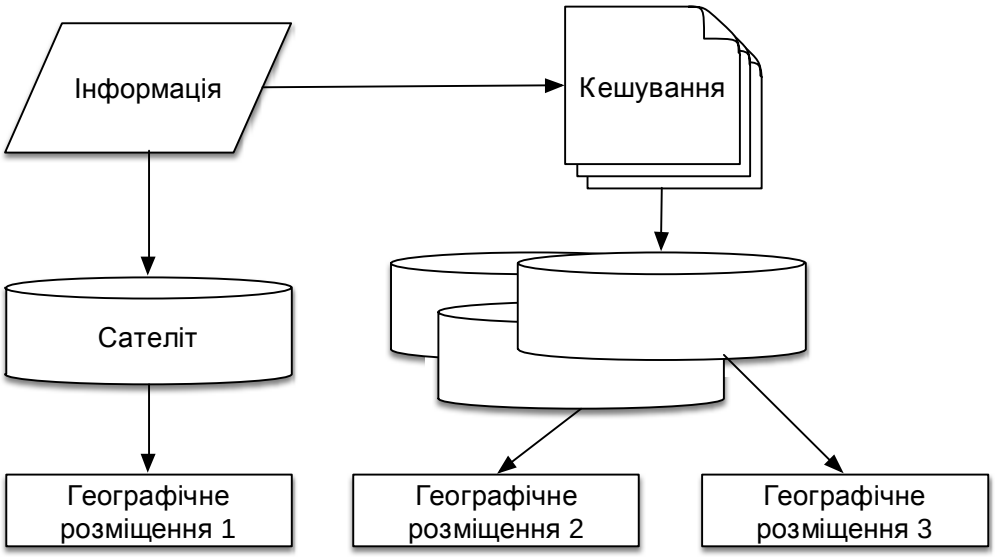

Рис. 4. Метод наближення даних до клієнта

Зберіганням та обробкою інформації при даному методі займається основний сервер. Основні задача сателітів забезпечити обмін інформацією з користувачем таким чином, щоб він виявився якомога “ближче” до споживача (рис. 4).

Крім самого факту, що інформація буде доставлена користувачеві чи прийнята від нього, потрібно забезпечити якість доставки. Це передбачає максимально можливу швидкість обміну цією інформацією.

Метод також передбачає можливість “наблизити” інформацію до користувача географічно. Це пов'язано з пропускною здатністю каналів (відсутністю іноді хороших магістральних каналів), а також з різницею у вартості локального і зовнішнього трафіку для кінцевого користувача (наприклад, у різних регіонах).

Таким чином, усвідомивши необхідність географічного розподілення для інформації, встановлюють серверни-

сателіти в безпосередній близькості від споживачів.

Абстрактний алгоритм наближення даних до клієнта, у якому беруть участь діф сутності - основний сервер і користувач, полягає в наступному:

•1. Визначення розміщення користувача. Сподіватися, що він сам розповість цю інформацію про себе, не доводиться, тому беруть *IP*-адресу користувача і виконують пошук в базі даних *GeoIP* (яких сьогодні досить багато, як платних, так і безкоштовних), і отримують на виході інформацію про місцезнаходження користувача: його країну, регіон, місто, назва його провайдера.

2. Визначення розміщення інформації користувача. Ця інформація розташована завжди на основному сервері, а сервер має своє фізичне місце розташування, заздалегідь відоме. Тому інформація на сервері теж має своє географічне місце розташування: ті ж країна, регіон, місто і т.п. Крім того, при бажанні користувача, можна створювати копії інформації на інших основних серверах, таким чином одна й та ж інформація буде доступна на декількох основних серверах, а значить, в кількох географічних точках.

3. Визначення множини сателітів для доступу до інформації. Логічно було б обчислити відстань від користувача до інформації і всіх сателітів, що забезпечать до неї доступ, і вибрати самий близький до користувача сателіт. Ця відстань не завжди збігається з відстанню на карті між двома точками, і є швидше мірою якості і пропускної спроможності каналів між регіонами, країнами або містами (або навіть між окремими провайдерами).

4. Вибір потенційного сателіта для доступу до інформації. У результаті виконання попереднього кроку отримують зважений орієнтований граф, на якому необхідно вирішити задачу пошуку найкоротшого шляху (мінімізація суми ваг ребер графа, що входять в цей шлях). Це класична задача, яка може бути легко вирішена за поліноміальний час.

5. Пошук оптимального шляху. Класичні алгоритми пошуку найкоротшого шляху на графах при практичній реалізації для цієї задачі виявилися досить громіздкими і вимагали достатньо багато часу, що є недопустимим. Потрібно враховувати специфіку утвореного графа. Граф хоч і є слабозв'язаний, але все-таки його розміри досить великі, тому в реальному часі для кожного відвідувача виконувати такий розрахунок не є найбільш ефективним рішенням. Відомо, що при всіх пошуках найкоротшого шляху кінцевим пунктом шляху будуть місця, де розташовані сателіти, таким чином, число різних кінцевих точок шуканого шляху фіксоване і відносно невелике. Тепер достатньо розрахувати і закешувати довжини найкоротших шляхів від всіх вершин графа (де можуть розташовуватися користувачі) до місць розташування сателітів.

6. Збереження оптимального шляху в базі клієнтів. Уже оброблена інформація буде використана в реальному часі надалі, коли за інформацією звернеться користувач. Тому завжди необхідно виконувати поновлення бази даних клієнтів.

Але розрахована один раз і збережена інформацію про оптимальні шляхи не є динамічною, що в реальних умовах роботи відобразиться на якості надання послуг. Тому такий підхід зі статично розрахованими шляхами не є добрим вирішення.

Тому, пропонується модифікований метод, який крім переваг цього підходу до пошуку мінімального шляху за географією використовує ще два додаткові підходи:

1. На основі даних про мережеву топологію:

•*BGP*-маршрути в режимі реального часу;

- База даних *RIPE*;
- Таблиці локальних адресів регіональних провайдерів;

2. За інтегральними метриками:

- мережева затримка;
- кількість хопів;

• *AS* (автономних систем) на шляху.

У результаті метод вибору шлюзу за складністю запиту можна зобразити у фігляді діаграми потоків даних (рис. 5).

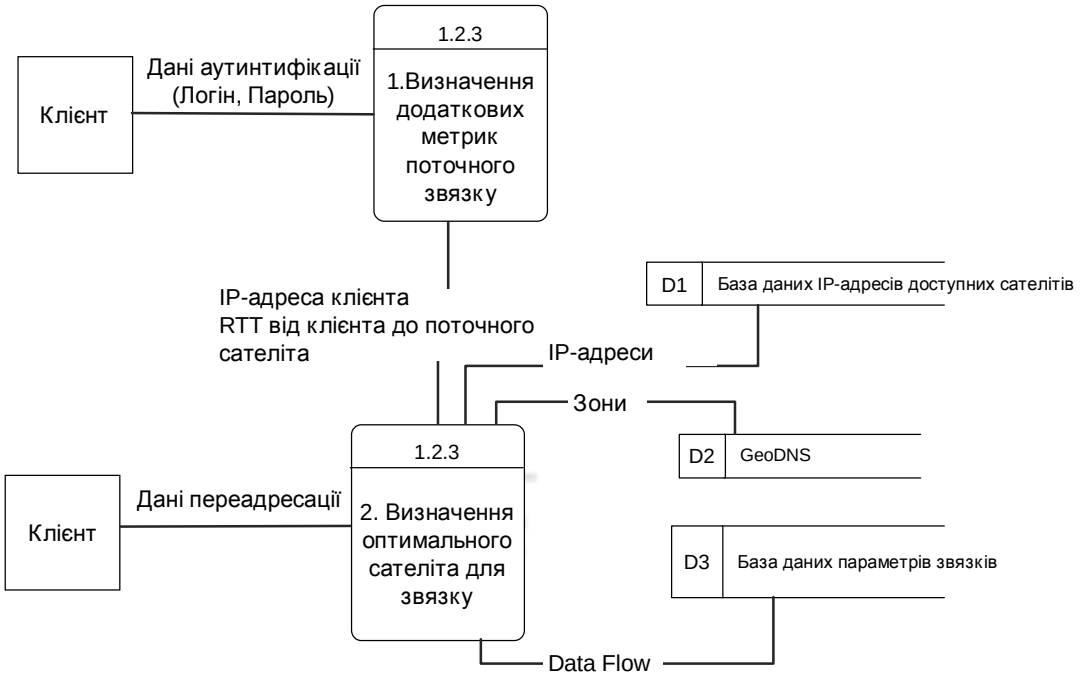

Рис 5. Діаграма потоків даних методу вибору шлюзу доступу

Висновки

Практичне використання даного методу в програмній реалізації дозволило отримати систему дистрибуції даних до кінцевого користувача, з максимально швидким доступом до даних. Хоча й запропонований метод ще не містить достатнього наукового підгрунтя, але враховуючи його базування на практичних основах і підтвердженість у реальних системах дистрибуції дозволяють його використання.

Список літератури

1. AGARWALA, Sandip, et al. iCostale: Adaptive cost optimization for storage clouds. In: Cloud Computing (CLOUD), 2011 IEEE International Conference on. IEEE, 2011. P. 436-443.

2. Garay, Juan Alberto, et al. Method and apparatus for the secure distributed storage and retrieval of information. U.S. Patent No 6,192,472, 2001.

3. Salama, Hussein Farouk, David R. Oran, and Dhaval N. Shah. "Method and apparatus for automatic inter-domain routing of calls." U.S. Patent No. 6,584,093. 24 Jun. 2003.

4. Vardi, Yosef, et al. "System and method for mapping wireless access points." U.S. Patent No. 8,126,476. 28 Feb. 2012.

5. Zhang, Qi, Lu Cheng, and Raouf Boutaba. "Cloud computing: state-of-the-art and research challenges." Journal of internet services and applications 1.1 (2010): P. 7-18.

Статтю подано до редакції 22.09.2015